

This listing of claims will replace all prior versions and listings of claims in the application.

**Listing of Claims**

1. (Currently Amended) A computer-readable memory medium, comprising:

a first software component adapted to create a graphical representation of an object embodied as code within the software component, wherein the code, ~~when rendered upon a display screen as said graphical representation,~~ comprises text and other displayable content;

an application program ~~adapted to create the graphical representation by running the application program under an operating system;~~ and

a second software component ~~is adapted for drawing the text, and wherein the first software component is invoked during runtime by the application program to define the visual attributes of the text, but not to draw the text, and wherein the second software component is invoked to draw the text, using only said text using the visual attributes.~~

2. (Currently Amended) The memory as recited in claim 1, wherein the operating system ~~assigns~~ places the text ~~drawing placed to be drawn~~ in a buffer, and wherein the first software component ~~enables~~ the buffered text to be edited prior to being drawn.

3. (Currently Amended) The memory as recited in claim 1, wherein the first software component ~~may be~~ is used ~~to-for~~ drawing the graphical representation of the object on a display screen, and wherein the second software component is used for drawing the text upon the graphical representation of the object.

4. (Currently Amended) The memory as recited in claim 1, wherein the first software component is adapted to support undo and redo editing of the text content in ~~drawn upon~~ the graphical representation of the object.

5. (Original) The memory as recited in claim 1, wherein the object is part of a graphical user interface associated with the application program.

6. (Original) The memory as recited in claim 1, wherein the application program is written in Java programming language.

7. (Currently Amended) The memory as recited in claim 1, wherein the first and second software components comprises a Java virtual machine of a Swing application program interface.

8. (Currently Amended) A method for drawing an object embedded within software code, said object ~~comprises comprising~~ text and other displayable content created by an application program running under an operating system, the method comprising:

executing a first software component to create a graphical representation of the object to define the visual attributes of the object ~~absent creation of any~~ without creating the text attributable to the object; and

executing a second software component to draw the text attributable to the object by using the visual attributes to fetch code that is independent of the operating system.

9. (Currently Amended) The method as recited in claim 8, wherein said executing a first software component comprises creating a label upon the display ~~absent any text~~.

10. (Currently Amended) The method as recited in claim 8, wherein said executing a first software component comprises creating a border upon the display ~~absent any text~~ within the border.

11. (Currently Amended) The method as recited in claim 8, further comprising the operating system ~~assigning placing the text drawing placed to be drawn~~ in a buffer, and wherein said executing a first software component comprises editing the buffered text ~~edited prior to the buffered text~~ being drawn.

12. (Currently Amended) The method as recited in claim 8, further comprising using the first software component to draw the graphical representation of the object on a display screen, and using the second software component to draw the text upon the graphical representation of the object.

13. (Currently Amended) The method as recited in claim 8~~12~~, further comprising the first software component supporting undo and redo editing of the text ~~content in drawn upon~~ the graphical representation of the object.

14. (Original) The method as recited in claim 8, wherein the object is part of a graphical user interface associated with the application program.

15. (Currently Amended) The method as recited in claim 8, ~~wherein the further comprising using the second software component to establish an appearance and behavior of the object is that are independent of the operating system.~~

16. (Currently Amended) A computer-readable storage device, comprising:

a windows-based operating system;

an application program running under the operating system;

an object and text associated with the object, both created at runtime by the application program, wherein the application program is adapted for;

invoking a first software component adapted ~~to create for creating~~ a graphical representation of the object ~~to define and defining the~~ visual attributes of the object, ~~but not to draw without~~ drawing the text; and

invoking a second software component adapted for ~~fast text drawing to draw the text, using said text using the~~ visual attributes.

17. (New) The memory as recited in claim 3, further comprising a peer component adapted for redirecting a memory call to invoke text drawing methods of the second software component rather than text drawing methods of the first software component.